SHARP APL Reference Card

For assistance please contact your nearest office.

ABERDEEN	(0224)25298
AMSTERDAM	(020)24 40 50
ATLANTA	(404)953-1020
BIRMINGHAM	(021)359-6964
BOSTON	(617)523-2506
BRUSSELS	(02)649 99 77
CALGARY	(403)265-7730
CHICAGO	(312)648-1730
CLEVELAND	(216)261-0800
COPENHAGEN	(01)112 434
DALLAS	(214)369-1131
DUESSELDORF	(0211)57 50 16
EDMONTON	(403)428-6744
GLOUCESTER	0452 28106
HOUSTON	(713)526-5275
LONDON, CANADA	(519)434-2426
LONDON, ENGLAND	(01)730-0361
MELBOURNE	(03)244-417
MIAMI LAKES	(305)556-0577
MINNEAPOLIS	(612)854-3405
MONTREAL	(514)866-4981
NEW YORK CITY	(212)986-3366
NEWPORT BEACH	(714)644-5112
OTTAWA	(613)236-9942
PALO ALTO	(415)327-1700
PHILADELPHIA	(215)735-3327
RESEDA	(213)343-4617
ROCHESTER	(716)546-7270
SAN FRANCISCO	(415)673-4930
SEATTLE	(206)453-1661
STOCKHOLM	(08)21 10 19
SYDNEY, AUSTRALIA	(02)232-5914
TORONTO	(416)364-5361
VANCOUVER	(604)682-7158
VICTORIA	(604)388-6365
VIENNA	(222)576571
WARRINGTON	(0925)50413/4
WASHINGTON	(202)293-2915
WINNIPEG	(204)947-1241
ZURICH	241 52 42
SHARP APL OPERATOR	(416)363-2051

SCALAR DYADIC FUNCTIONS

X+Y		X plus Y
X-Y		X minus Y
X×Y		X times Y
X÷Y		X divided by Y
X⋆Y		X to the Y-th power
X⌈Y		Maximum of X and Y
X⌊Y		Minimum of X and Y
X|Y		X-residue of Y
X⍟Y		Base-X logarithm of Y
X!Y		Binomial coefficient: Y items taken X at a time
X○Y		See trigonometric functions
X<Y	*	X less than Y
X≤Y	*	X less than or equal to Y
X=Y	*	X equal to Y
X≥Y	*	X greater than or equal to Y
X>Y	*	X greater than Y
X≠Y	*	X not equal to Y
X∧Y		X and Y
X∨Y		X or Y
X⍲Y		Not both X and Y
X⍱Y		Neither X nor Y

TRIGONOMETRIC FUNCTIONS

R←X○Y

(Y in radians)		(R in radians)	
X	R	X	R
0	(1-Y⋆2)⋆.5		
1	Sine Y	¯1	Arcsin Y
2	Cosine Y	¯2	Arccos Y
3	Tangent Y	¯3	Arctan Y
4	(1+Y⋆2)⋆.5	¯4	(¯1+Y⋆2)⋆.5
5	Sinh Y	¯5	Arcsinh Y
6	Cosh Y	¯6	Arccosh Y
7	Tanh Y	¯7	Arctanh Y

SCALAR MONADIC FUNCTIONS

+Y		Y
-Y		0-Y
×Y		Signum Y
÷Y		Reciprocal of Y
⋆Y		e to the Y-th power
⌈Y	*	Ceiling of Y
⌊Y	*	Floor of Y
|Y		Absolute value of Y
⍟Y		Natural logarithm of Y
!Y		Factorial Y; Gamma Y+1
○Y		Pi times Y
?Y	#	A random number from ⍳Y
~Y		Not Y

* ⎕CT dependent
\# ⎕IO dependent

SYMBOLS

( )	Parentheses for nesting
[ ]	Brackets for indexing
⎕	Quad for input-output
⍞	Quote-quad for character input
'	Quote delimits character literals
⍝	Lamp indicates comment
¯	Negative sign
E	Exponential notation
Δ	Delta. Trace (TΔ) and stop (SΔ) control
⌶	I-beam for system functions
.	Decimal point
^	Caret locates errors
;	Semicolon separates list elements
◇	Diamond used as statement separator
→	Branch
:	Colon delimits labels and locks
⌷	Squish-quad display of non-printable characters

The following characters are available as graphics:

$ α { ⊢ ⊂ ∩ ¨
¢ ω } ⊣ ⊃ ∪ _

FUNCTION DEFINITION

Function definition is opened or closed by ∇. Use of ⍫ at the close will lock the function. The following are valid with open definition.

[⎕]	Display entire function
[**n**⎕]	Display line **n**
[⎕**n**]	Display from line **n** to the end
[**n**⎕**p**]	Display line **n** and position at **p** prior to editing
[**n**⎕0]	Display line **n** and position at right-hand end of line
[**n**]	(Re)define line **n**

MIXED FUNCTIONS

X⍴Y	Reshape Y to have dimensions of X
⍴Y	Dimensions of Y
X[Y]	#Y-th elements of X
X⍳Y	*#First locations of Y within vector X
⍳Y	#Y consecutive integers from origin (0 or 1)
X∊Y	* Membership of X in Y
X⊤Y	Representation of Y in number system X
X⊥Y	Value of the representation Y in number system X
X?Y	#X integers selected randomly without repetition from ⍳Y
X⌽[Z]Y	Rotation by X along the Z-th dimension of Y
⌽[Z]Y	Reversal along the Z-th dimension of Y
X⍉Y	#Transpose of Y according to X
⍉Y	Transpose of Y (⌽⍳⍴⍴Y)⍉Y
X,[Z]Y	Catenate or laminate Y to X along Z-th dimension
,Y	Ravel of Y (makes Y a vector)
X↑Y	Take first or last X elements of Y as X is + or -
X↓Y	Drop first or last X elements of Y as X is + or -
X←Y	X is assigned the value of Y
⍋Y	#Index vector such that Y[⍋Y] is in ascending order
⍒Y	#Index vector such that Y[⍒Y] is in descending order
X/[Z]Y	X (logical) compressing along the Z-th dimension of Y
X\\[Z]Y	X (logical) expanding along the Z-th dimension of Y
R←X⌹Y	Matrix divide. X approximates R+.×Y
⌹Y	Generalized matrix inverse of Y
⍎Y	Executes the character string Y
⍕Y	Character representation of Y
X⍕Y	Numeric expression Y is converted to character according to numeric format control vector X

⊛ below denotes any scalar dyadic function

⊛\[Z]Y	⊛ applied cumulatively along the Z-th dimension of Y
⊛/[Z]Y	The ⊛ reduction along the Z-th dimension of Y
X∘.⊛Y	Generalized outer product of X and Y
X⊛.⊛Y	Generalized inner product of X and Y e.g. X+.×Y is ordinary matrix product of X and Y

⊖	⌽ applied to first dimension (reverse or rotate)
⌿	/ applied to first dimension (compress or reduce)
⍀	\ applied to first dimension (expand or scan)

If the expression [Z] is omitted, the operation applies to the last dimension of the argument array Y. The expression [Z] is ⎕IO dependent.

* ⎕CT dependent
\# ⎕IO dependent

FILE SYSTEM FUNCTIONS

Function	Syntax	Permission
⎕APPEND	EXPR ⎕APPEND FN,PN	8
⎕APPENDR	R←EXPR ⎕APPENDR FN,PN	16384
⎕AVAIL	R←⎕AVAIL ⍳0	
⎕CREATE	FNM ⎕CREATE FN,PN	
⎕DROP	⎕DROP FN,N,PN	32
⎕ERASE	FNM ⎕ERASE FN,PN	4
⎕FHOLD	⎕FHOLD SVM	2048
⎕HOLD	⎕HOLD FN(S)	64
⎕LIB	R←⎕LIB LIB	
⎕NAMES	R←⎕NAMES	
⎕NUMS	R←⎕NUMS	
⎕RDAC	R←⎕RDAC FN,PN	4096,256
⎕RDCI	R←⎕RDCI FN,CN,PN	512
⎕READ	R←⎕READ FN,CN,PN	1
⎕RENAME	FNM ⎕RENAME FN,PN	128
⎕REPLACE	EXPR ⎕REPLACE FN,CN,PN	16
⎕RESIZE	P ⎕RESIZE FN,PN	1024
⎕SIZE	R←⎕SIZE FN,PN	1
⎕STAC	MAT ⎕STAC FN,PN	8192,256
⎕STIE	FNM ⎕STIE FN,PN	
⎕TIE	FNM ⎕TIE FN,PN	2
⎕UNTIE	⎕UNTIE FN(S)	

CN	Component number
EXPR	Data
FN	File number

FILE SYSTEM FUNCTIONS

FNM	File name
LIB	Library number
MAT	3-column access matrix
N	Integer
P	Positive integer
PN	Passnumber or ⍳0
R	Result
SVM	Scalar, vector, or 2-row matrix of integers

The syntax for *FNM* of ⎕*CREATE* and ⎕*RENAME* is **'** [library-number] filename [file-size-limit] **'** where [. . .] is optional.

Complete access to a file can be given by a permission code of ¯1.

REPORT FORMATTING

The following is a guide to the composition of the left and right arguments of the formatting function ⎕*FMT*:

'fp,fp, . . . ,fp' ⎕*FMT* **(expr;expr; . . . ;expr)**

expr Scalar, vector, or matrix

fp Format phrase

m*A***w**	Character data
m*E***w.d**	Floating-point
mq*F***w.d**	Fixed-point
mq*I***w**	Integer
m*X***w**	Skip
m*T***w**	Tab
<text>	Literal data
mq*Q*<pattern>	Controlled format

m Repetition factor (optional)

w Field width

d Number of decimal positions

q Qualifiers (optional)

B	Blank if zero
C	Comma insertion
*K***e**	Scaling by 10⋆**e**
L	Left justify
Z	Zero fill
M<text>	Left of negative result
N<text>	Right of negative result
P<text>	Left of non-negative result
Q<text>	Right of non-negative result
R<text>	Background for result
S<text>	Standard symbol substitution

<	<⊂⍞⍬	Left text delimiter
>	>⊃⍞⍬	Matching right text delimiter

SELECTED PUBLIC WORKSPACES

1 *ARBOUT*	Tables of ASCII and BCD transmission codes
1 *BTASKREQ*	B-task submission
1 *DEMOS*	Documentation of SHARP APL demos
1 *FILESORT*	File sorting utility
1 *FORMAT*	Functions complementary to ⎕*FMT*
1 *HSPRINT*	Remote highspeed print
1 *NETWORK*	Documents communication network
1 *NEWS*	System enhancements, schedules, phone numbers
1 *PUBLICLIBS*	Locates objects in public libraries
1 *REFERENCE*	On-line reference card
1 *SATN*	SHARP APL Technical Notes - on-line
1 *USAGE*	Accounting information by month
1 *WSFNS*	Workspace control functions
3 *PLOT*	Plotting functions
4 *EDIT*	Text editor
5 *FONT*	Each APL character and its English name
5 *TERM*	Complete documentation of)*TERM*
6 *WSTRANSFER*	Source-format WS interchange
7 *WSDOC*	Workspace documenter

SHARP APL TECHNICAL NOTES

SATN-0	1 *JAN* 76		*SATN INTRODUCTION*
SATN-1	1 *JAN* 76		*TASKID*
SATN-2	15 *SEP* 77	*REV*: 2	*CONTROL MESSAGES*
SATN-3	1 *JAN* 76		⎕*OUT*
SATN-4	1 *APR* 78	*REV*: 2	*N-TASKS AND B-TASKS*
SATN-5	15 *JUL* 76	*REV*: 1	*BATCH APL*
SATN-6	1 *JAN* 76		*EXECUTE*
SATN-7	1 *JAN* 76		*LATENT EXPRESSION*
SATN-8	15 *AUG* 76	*REV*: 1	*HSPRINT*

SHARP APL TECHNICAL NOTES

SATN-9	1 *JUN* 76		*USAGE INQUIRY SYSTEM*
SATN-10	1 *JAN* 77	*REV*: 1	*SORTREQ*
SATN-11	1 *JAN* 76		)*RESET*
SATN-12	1 *JAN* 76		)*COPY*
SATN-13	10 *MAR* 78		*EARLY WARNINGS*
SATN-14	15 *JAN* 77		*PACKAGE - A NEW VARIABLE TYPE*
SATN-15	20 *APR* 76		*INDEX*
SATN-16	20 *APR* 76		*FILE SYSTEM MUST-WRITE BUFFERS*
SATN-17	15 *JUL* 76		*FORMATTING PRIMITIVE*
SATN-18	1 *JUL* 76		⎕*FMT*
SATN-19	1 *JAN* 77		*FILEPRINT*
SATN-20	1 *JUN* 78	*REV*: 3	*SYSTEM VARIABLES, SESSION VARIABLES AND SYSTEM FUNCTIONS*
SATN-21	15 *JAN* 77		⎕*WS AND* ⎕*FD*
SATN-22	15 *JAN* 77		*APL WORKSPACE TRANSFER*
SATN-23	7 *JUN* 77		*COMPARISON TOLERANCE*
SATN-24	25 *MAR* 77		)*SYMBOLS*
SATN-25	15 *MAY* 77		*EXTENSIONS TO ARGUMENT PASSING*
SATN-26	10 *SEP* 77		*ENHANCEMENTS TO THE FILE SYSTEM*
SATN-28	11 *JUL* 77		*TERMINAL CONTROL*

SYSTEM COMMANDS

Note: Bracketed words are optional.

) **[n]**	Allow editing of immediate execution line at position **n**
)**nnnnnnn [:lock]**	Sign-on - invalid sign-on acts as)*BLOT*
)*BLOT*	Print a mask for typing secure information
)*CLEAR*	Clear the active workspace
)*CONTINUE* [*HOLD*] **[:lock]**	Terminate a session and store the active workspace in *CONTINUE*
)*COPY* **name [:lock] [list]**	Copy objects from a workspace
)*DROP* **name [:lock]**	Delete a stored workspace
)*ERASE* **list**	Erase objects
)*FNS* **[letters]**	List names of functions
)*GROUP* **name list**	Define a group
)*GROUP* **name**	Disperse a group
)*GRP* **name**	List members of a group
)*GRPS* **[letters]**	List names of groups
)*KEYB LOCK*	Lock keyboard (to receive messages)
)*KEYB NOMSG*	Suppress incoming messages
)*KEYB*	Return keyboard to normal state
)*LIB* **[nnn]**	List names of workspaces in library
)*LOAD* **name [:lock]**	Activate a copy of a stored workspace
)*MSG* **taskid text**	Send message to designated terminal
)*MSGN* **taskid text**	Send message to designated terminal without locking sender's keyboard to receive reply
)*OFF* [*HOLD*] **[:lock]**	Terminate a work session
)*OPR* **text**	Send message to SHARP APL operator
)*OPRN* **text**	Send message to SHARP APL operator without locking sender's keyboard to receive reply
)*PCOPY* **name [:lock] [list]**	As)*COPY*, but protect contents of active WS from being overwritten.
)*PORTS* **aaa**	List port number and task-ID of user **aaa**
)*PORTS* **nnnnnnn**	List port number and task-ID of user **nnnnnnn**
)*RESET*	Clear the)*SI* stack
)*SAVE* **[name] [:lock]**	Store a copy of the active workspace
)*SEAL*	Lock all functions in the workspace; prevent dispersal of contents
)*SI*	Display the state indicator
)*SIV*	Display state indicator and names of local variables
)*SYMBOLS* **[n]**	Display/set symbol table size
)*TERM* **[name]**	Display/set terminal type. See WS 5 *TERM* for current list of **names**
)*VARS* **[letters]**	List names of variables
)*WSID* **[name]**	Display/set workspace name

VALID ARGUMENTS TO TERM COMMAND:

AJ832 ASCII DCTTAPE DCTTTYS GET1200 HP2641
IBM2741 MRX1240 REALTTY SDATA TEK4013 TY33

N-TASKS AND B-TASKS

R←⎕*RUN* **'signon loadwsid savewsid cpulim conlim'** Initiates an N-task. **signon** and **loadwsid** may each be optionally elided by the character ':'.

R

0	0	Split-ws new task
0	*N*	Successful *N*=taskid
1	*N*	Invalid argument at character *N*
2	0	System resources exhausted
3	0	User N-task quota (4) exceeded
4	0	Ws quota exhausted
5	0	Ws full in **loadwsid**
6	0	Symbol table full in **loadwsid**
10	0	Sign-on number not in system

N-TASKS AND B-TASKS

```
11 0 Sign-on number locked out
12 0 Loadwsid not found
13 0 Loadwsid invalid ⎕LX
14 0 System resources exhausted
15 0 User may not use ⎕RUN
16 0 Hardware error
```

R←⎕*RUNS* — Task information in matrix format. Columns are:
taskid user initiator termtype cpu con cpulim conlim

R←⎕*BOUNCE* **taskid** — Terminates tasks specified by **taskid.** Task executing ⎕*BOUNCE* must be **user** or **initiator** to be successful. Result is Boolean indicating successful terminations.

R←⎕*OUT* **pr,fn** — Sets/displays mode in which data is printed. ⎕*OUT* ⍳0 returns the current setting without modification. Otherwise the previous setting is returned and the mode set as follows:
Print data at terminal if **pr** is 1.
Append data to file number **fn** if **fn** is not zero. If **fn** is zero do not append data to file.

B-tasks — Requests are submitted via the functions in WS 1 *BTASKREQ*

SYSTEM VARIABLES

⎕*CT* — Comparison tolerance (i.e. Fuzz). Maximum relative difference between two floating point numbers considered equal by SHARP APL. Default is approx. 1E¯14.

⎕*IO* — Index origin, can be set 0 or 1. Default is 1.

⎕*LX* — Latent expression executed upon WS load. Default is ''.

⎕*PP* — Print precision. Number of digits printed when floating point numbers are displayed. Default is 10.

⎕*PW* — Page width. Default is 132.

⎕*RL* — Random link (i.e. Seed). Default is 16807.

SESSION VARIABLES

⎕*HT* — Horizontal tab positions. Default is ⍳0.

⎕*SP* — Session parameter. Default is ''.

SYSTEM FUNCTIONS

Function	Syntax	Definition
⎕*AI*	*R*←⎕*AI*	Returns a vector of accounting information. Element — Contents 1 — User number 2 — CPU time this session 3 — Connect time this session 4 — Keying time this session 5 — Characters this session 6 — Cumulative CPU time month-to-date 7 — Cumulative connect time month-to-date 8 — Zero (unused) 9 — Cumulative characters month-to-date ⎕*AI* is subject to extension. Do not assume a specific length.
⎕*ARBIN*	*R*←⎕*ARBIN X*	Arbitrary input. *X* is terminal-dependent transmission codes presented to terminal. *R* is numeric representation of user's response.
⎕*ARBOUT*	⎕*ARBOUT X*	Arbitrary output. *X* is terminal-dependent transmission codes presented to terminal.
⎕*AV*	*R*←⎕*AV*	Atomic vector. Vector of all 256 unique characters in the order of their internal representation.
⎕*CR*	*R*←⎕*CR NL*	Character matrix representation of the function named in *NL*.
⎕*DL*	*R*←⎕*DL X*	Delay. Causes at least *X*⌈10 seconds of delay prior to resuming execution. *R* is actual delay in seconds.
⎕*EX*	*R*←⎕*EX X*	*X* is a namelist matrix of identifiers whose most local value is to be erased. *R* is a Boolean vector indicating successful erasure.
⎕*FD*	*R*←*I* ⎕*FD X*	Function definition. *I* specifies the particular function definition service. *X* is a character vector or array.

Function	Syntax	Definition
⎕*FD*	*R*←1 ⎕*FD NL*	Character vector representation of the function named in *NL*.
	R←2 ⎕*FD NL*	Character matrix representation of the function named in *NL*.
	R←3 ⎕*FD X*	*X* is a vector or matrix character representation of a function. If 3 ⎕*FD* is successful the function will become defined in the WS. *R* will be the function name. If unsuccessful *R* will be a two element numeric error report. Element 2 of *R* is the location within *X* that the error was discovered. Element 1 is documented below: *R*[1] — Error 1 — WS full 2 — Defn error 3 — Char error 4 — Symbol table full 5 — Unmatched quotes 6 — System error - notify your SHARP APL representative 7 — Empty line
	NL2←6 ⎕*FD NL1*	*NL1* is a namelist of identifiers whose most local value is to be erased. *NL2* is a namelist matrix of identifiers not so erased.
	NL2←7 ⎕*FD NL1*	*NL1* is a namelist of functions to be locked. *NL2* is a namelist matrix of the identifiers not so locked.
⎕*FX*	*R*←⎕*FX NL*	*NL* is a matrix character representation of a function. If ⎕*FX* is successful the function will become defined in the WS. *R* will be the function name. If unsuccessful *R* will be the origin-0 row index of the fault.
⎕*FI*	*R*←⎕*FI X*	Converts the character representation of vector of numbers into a numeric vector. Incorrectly formed numbers are converted to zero.
⎕*LC*	*R*←⎕*LC*	Line counter. Vector of line numbers corresponding to)*SI*.
⎕*LOAD*	⎕*LOAD X*	Loads the WS named in *X*.
⎕*NC*	*R*←⎕*NC NL*	Name classification of each name in the namelist matrix *NL*. *R* Class 0 Name is free 1 Label 2 Variable 3 Function 4 Other
⎕*NL*	*NL*←⎕*NL X*	Name list. *X* is a scalar or vector of classes 1 2 3 (See ⎕*NC*). The result is a namelist matrix of all identifiers of those classes in the WS.
	NL←*A* ⎕*NL X*	*A* is a scalar or vector of alphabetics. The namelist matrix returned is restricted to names beginning with letters in *A*.
⎕*QLOAD*	⎕*QLOAD X*	Loads the WS named in *X* without displaying savedate.
⎕*TS*	*R*←⎕*TS*	Timestamp. 7 element integer vector containing current year, month, day, hour, minute, second, and millisecond.
⎕*UL*	*R*←⎕*UL*	User load. Number of tasks signed on.
⎕*VI*	*R*←⎕*VI X*	Boolean vector indicating the correctly formed numbers in its character argument.
⎕*WA*	*R*←⎕*WA*	Workspace available. The number of bytes remaining available in the workspace.
⎕*WS*	*R*←*I* ⎕*WS X*	Workspace descriptions. *I* specifies the particular description required.
	NL←1 ⎕*WS X*	Namelist of certain classes of identifiers in the workspace. *X* Result 1 Functions 2 Variables 4 Groups 8 Labels
	R←2 ⎕*WS X*	*X* Result 1 Character workspace ID 2 Character matrix)*SI* 3 Workspace Environment Element — Contents 1 — ⎕*IO* 2 — ⎕*PP* 3 — ⎕*PW* 4 — Reserved 5 — Fuzz. ⎕*CT*×16⋆14 6 — ⎕*RL* 7 —)*SYMBOLS*

Function	Syntax	Definition
⎕*WS*		8 — No. symbols in use 9 — User number 10 — Port signed on 11 — Task-ID 12 — Translate set 0 N-task or B-task 1 Corr. 2741 2 BCD 2741 3 TY33 (64 char ASCII) 4 Typewriter pairing ASCII 5 Bit pairing ASCII 2 ⎕*WS* 3 is subject to extension. Do not assume a specific length. 4 Vector of three elements. Workspace size (no. bytes in use) Workspace owner number ⎕*RDCI*-style timestamp
	NL2←3 ⎕*WS NL1*	Character matrix of the names in the group designated by *NL1*.
	R←4 ⎕*WS NL*	Bytes consumed by the identifiers designated by *NL*.
	R←5 ⎕*WS NL*	)*SIV* information, one row per name in *X*, one column per)*SI* level, plus one for global environment. Value — Meaning ¯1 — Not localized at this level. 0 — Undefined (but localized) 1 — Function 2 — Variable 4 — Group 8 — Label
	R←6 ⎕*WS NL*	The value of the identifier designated by *NL*.

PACKAGE FUNCTIONS

A package is a variable; a heterogeneous collection of named objects.

Function	Syntax	Definition
⎕*PACK*	*P*←⎕*PACK NL* *P*←*NL* ⎕*PACK X*	Creates a package *P* from a namelist *NL*. Creates a package containing the object *X* with the name specified in *NL*.
⎕*PDEF*	⎕*PDEF P* *NL* ⎕*PDEF P*	Defines in the WS the objects in *P*. As above except only those objects named in *NL* are selected for definition.
⎕*PEX*	*P1*←*NL* ⎕*PSEL P2*	Package of objects selected from *P2* except those named in *NL*.
⎕*PINS*	*P1*←*P2* ⎕*PINS P3*	Inserts package *P3* into package *P2* to produce package *P1*.
⎕*PLOCK*	*P1*←⎕*PLOCK P2*	Package *P1* same as *P2* except all functions become locked.
	P1←*NL* ⎕*PLOCK P1*	As above but selected functions become locked.
⎕*PNAMES*	*NL*←⎕*PNAMES P*	Matrix of the names of the objects in package *P*. If *P* is not a package, returns ⍳0.
⎕*PNC*	*R*←*NL* ⎕*PNC P*	Package name class, analogous to ⎕*NC*. *R* Class ¯1 Valid name, not in package 0 Undef 2 Variable 3 Function 4 Ill-formed name
	R←⎕*PNC P*	Equivalent to *R*←(⎕*PNAMES P*)⎕*PNC P*.
⎕*PPDEF*	*NL2*←⎕*PPDEF P*	Defines in the WS the objects in *P* which were undefined in the WS. *NL2* contains names of objects not defined.
	NL2←*NL1* ⎕*PPDEF P*	As above except only those objects named in *NL1* are selected for definition.
⎕*PSEL*	*P1*←*NL* ⎕*PSEL P2*	Package of objects selected from *P1* named in *NL*.
⎕*PVAL*	*R*←*NL* ⎕*PVAL P*	The value of the object in *P* named in *NL*.

Printed in Canada
August 1978.